

Control and Observation in Distributed Environments

Warren Smith
Computer Sciences Corporation
NASA Ames Research Center

Motivation and Approach

- Large and distributed set of resources, services, applications
- There will be failures
- The grid must be managed
- Develop a general framework for observation and control
 - ◆ Observe and control a variety of resources and services
 - ◆ Support observation and control of user applications
- Extend the framework for specific tasks
 - ◆ Add components to observe new things
 - ◆ Add components to perform new actions
 - ◆ Add new logic for management

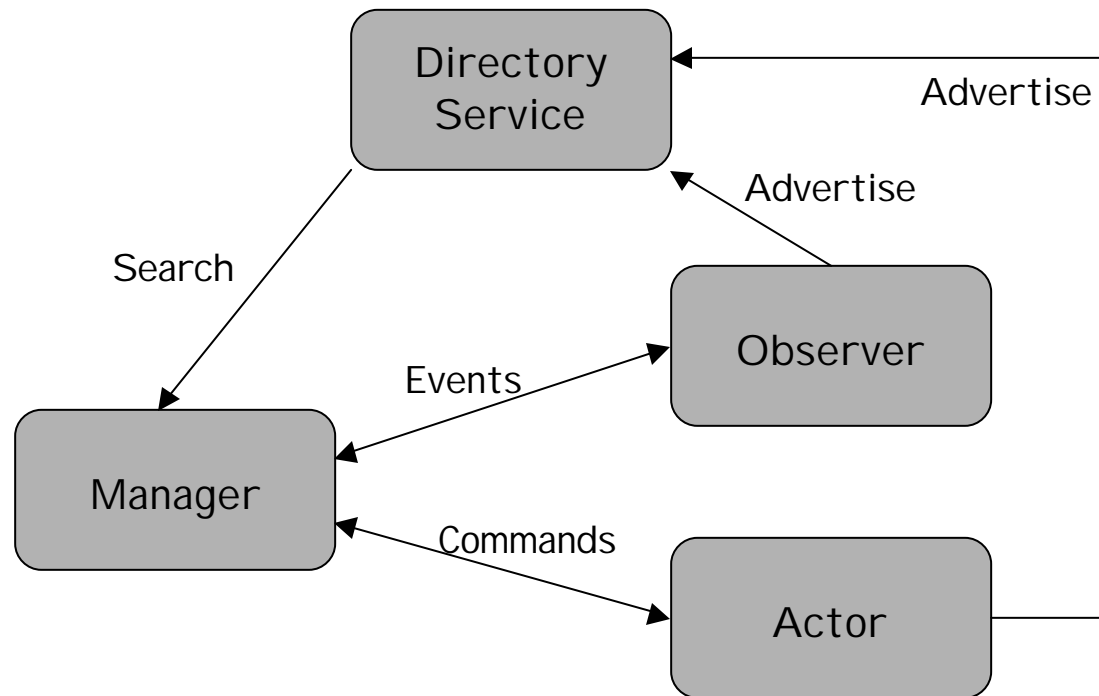
Why not use an existing system?

- No existing system met all our needs
 - ◆ Cannot be embedded in tools or applications
 - ◆ Limited fault management functionality
 - ◆ System- or application-specific information but not both
 - ◆ Lack of extensible data forwarding and gathering mechanisms
 - ◆ Incompatibility with security and authentication requirements of I PG
- Testbed for standards developed in the Grid Forum

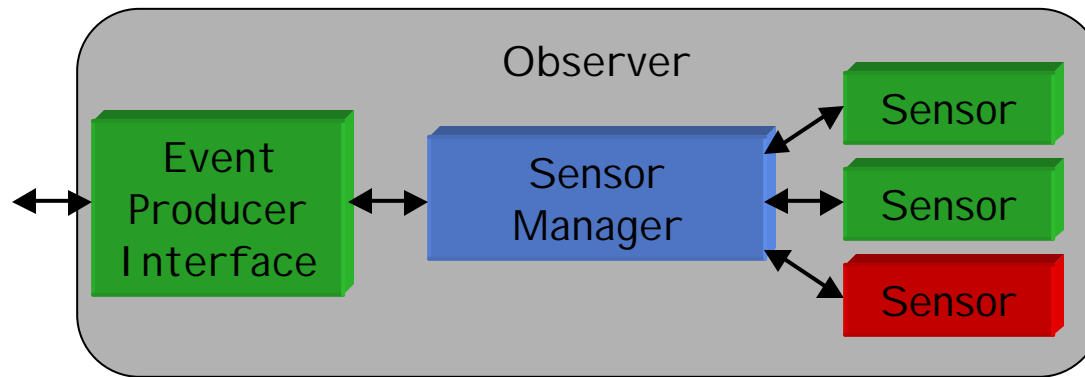
Goals of our Infrastructure

- Develop a general framework for observation and control
 - ◆ Observe and control a variety of resources, services, and applications
 - ◆ Scalable
 - ◆ Secure
- Framework should be extensible for specific tasks
 - ◆ Add new components for observing and performing actions
 - ◆ Easily add new logic for management
 - ◆ Modular
- Compatible with emerging standards
 - ◆ Grid Forum Performance Working Group
 - ◆ Grid Forum Event Service Working Group?

High-Level Architecture



Observer

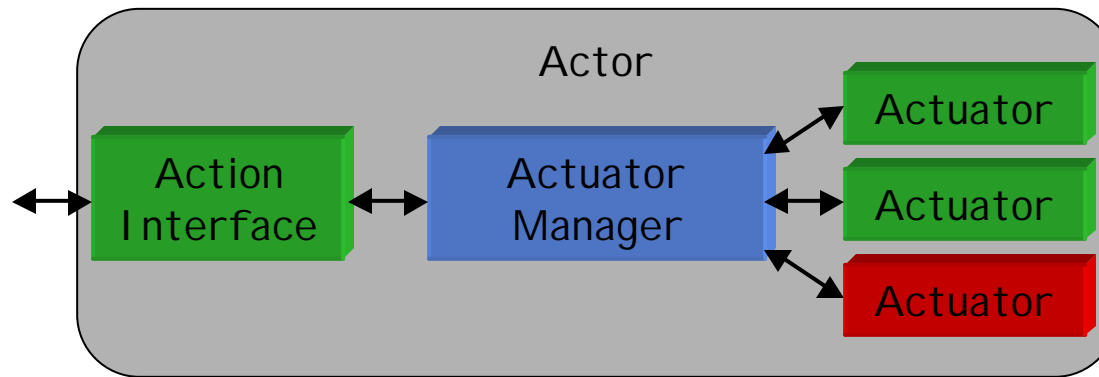


- Two low level components
 - ◆ Sensor
 - ◆ Event Producer Interface (Event Service)
 - Subscribe
 - Query
 - Available events
 - Event schemas
- Sensor manager
 - ◆ Manages sensors, subscriptions, and queries

Key



Actor

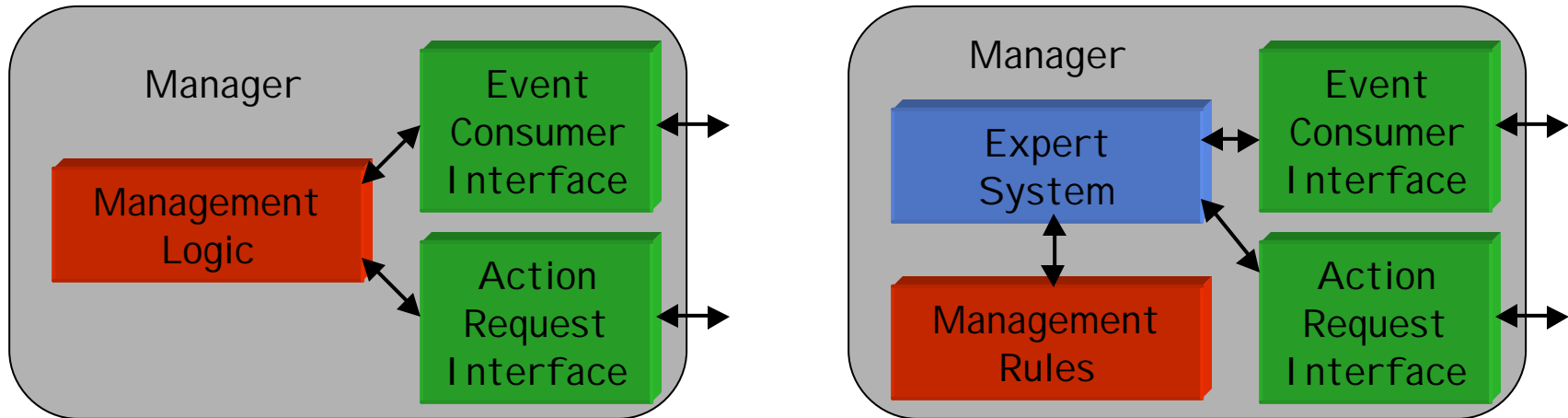


- Two low-level components
 - ◆ Actuator
 - ◆ Actor Server Interface
 - Request action (RPC)
 - Available actions
 - Action schemas
- Actuator Manager
 - ◆ Handles requests for actions

Key



Manager



- Two basic interface component
- 2 approaches to higher-level components
 - ◆ User writes management logic
 - ◆ User writes management rules and uses an expert system

Key



Directory Service

- Information about observers and actors
 - ◆ Contact location and protocol
 - ◆ Available events and actions
 - ◆ Who has access
- Event and action schemas
- Future: information about event consumers
 - ◆ Archives
 - ◆ Channels

Implementation

- Primary version written in C++
 - ◆ pthreads
 - ◆ CLIPS expert system
 - ◆ Communicates using TCP, UDP, or SSL
 - ◆ XML encoding of messages
 - ◆ expat parser
 - ◆ OpenSSL for authentication and security
 - Compatible with Globus Security Infrastructure identities
 - ◆ Currently runs under IRIX, Solaris, Linux
 - Manager code also in Java for GUIs
 - ◆ TCP, but no UDP or SSL yet
 - ◆ Xerces XML parser
 - Follows emerging Grid Forum standards
-

Standardization

- Performance Working Group of the Grid Forum
 - ◆ Architecture
 - ◆ Event representations
 - ◆ Directory service schema
 - ◆ Producer-consumer communication protocols
- Grid Event Service Working Group?
- This framework is compatible with the developing standards
- <http://www.gridforum.org>
- <http://www-didc.lbl.gov/GridPerf>

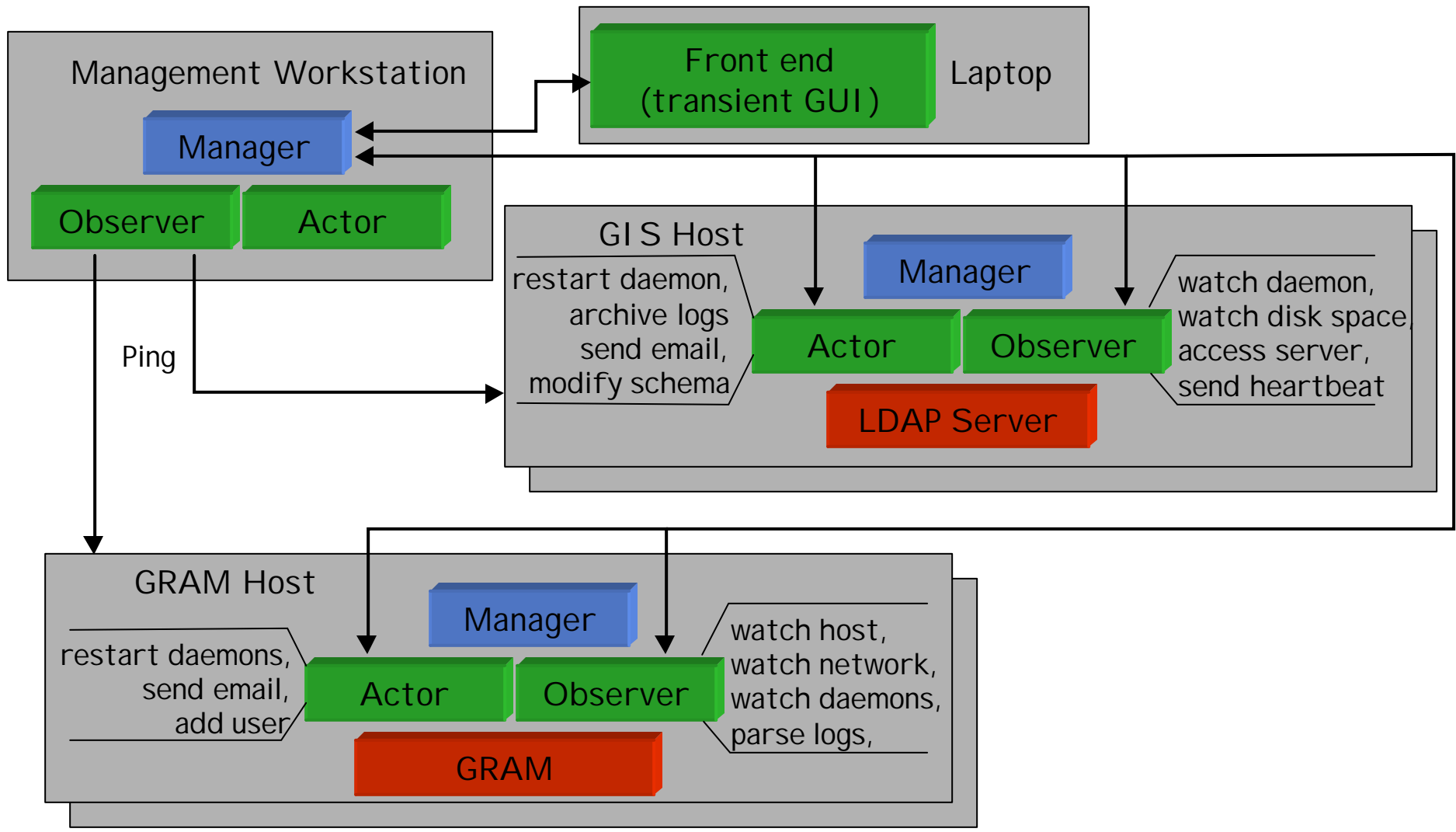
Uses of Infrastructure

- Management of a Globus-based computational grid
- Basis for an alternative Grid Information Service
- Grid accounting
- Application performance analysis
- Application steering

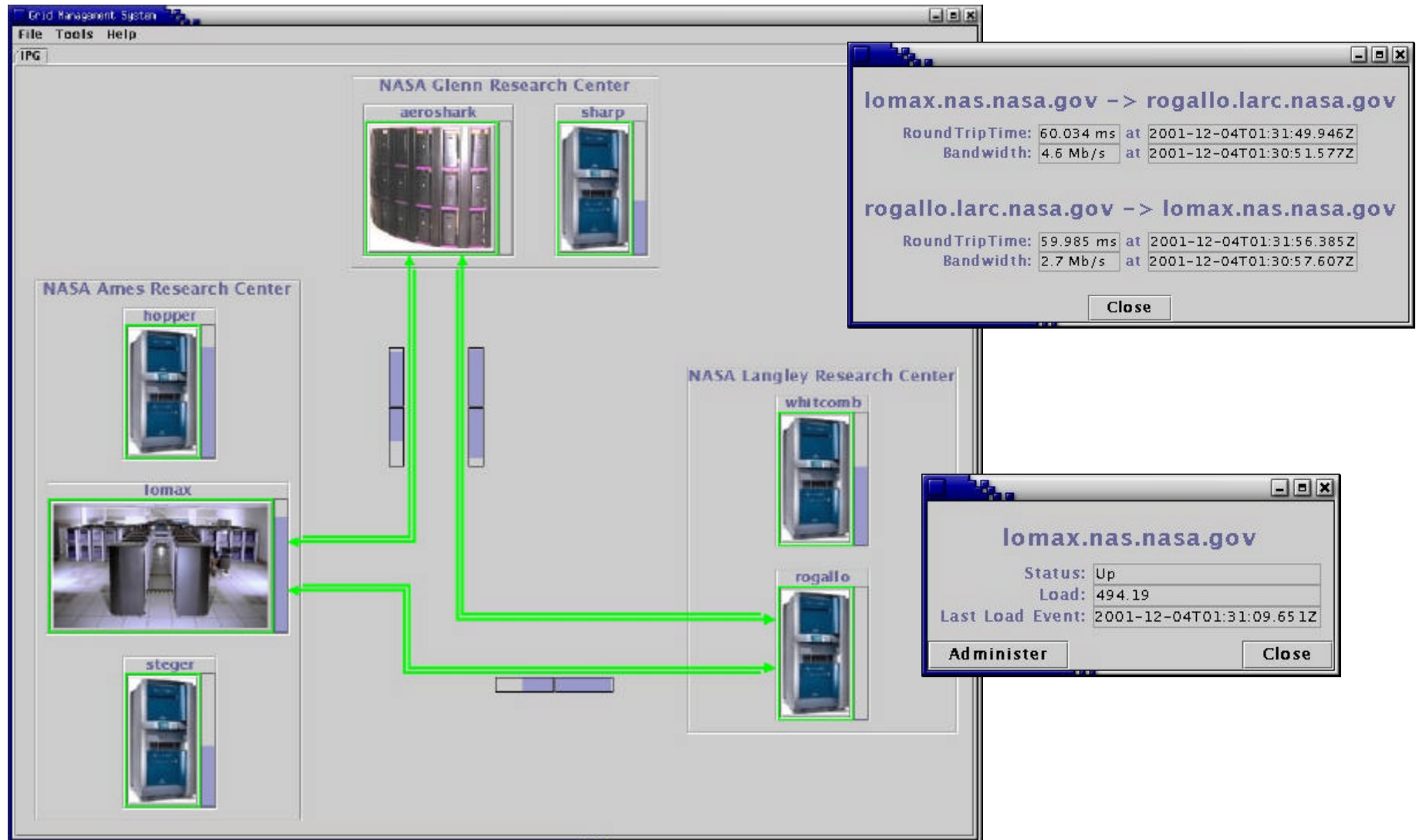
Grid Management System

- As grids get larger, it gets difficult to manage grids
- Things to observe:
 - ◆ Resource status and usage
 - ◆ Grid daemons
 - Grid Information Service servers
 - GRAM reporter daemons
 - ◆ Log files
 - Gatekeeper log
 - GRAM reporter log
- Things to control
 - ◆ Restarting/configuring GIS servers
 - ◆ Restarting GRAM daemons
 - ◆ Add/remove user mappings in grid-mapfiles
 - ◆ Add/remove certificate authorities
- Provide a nice GUI to do all this

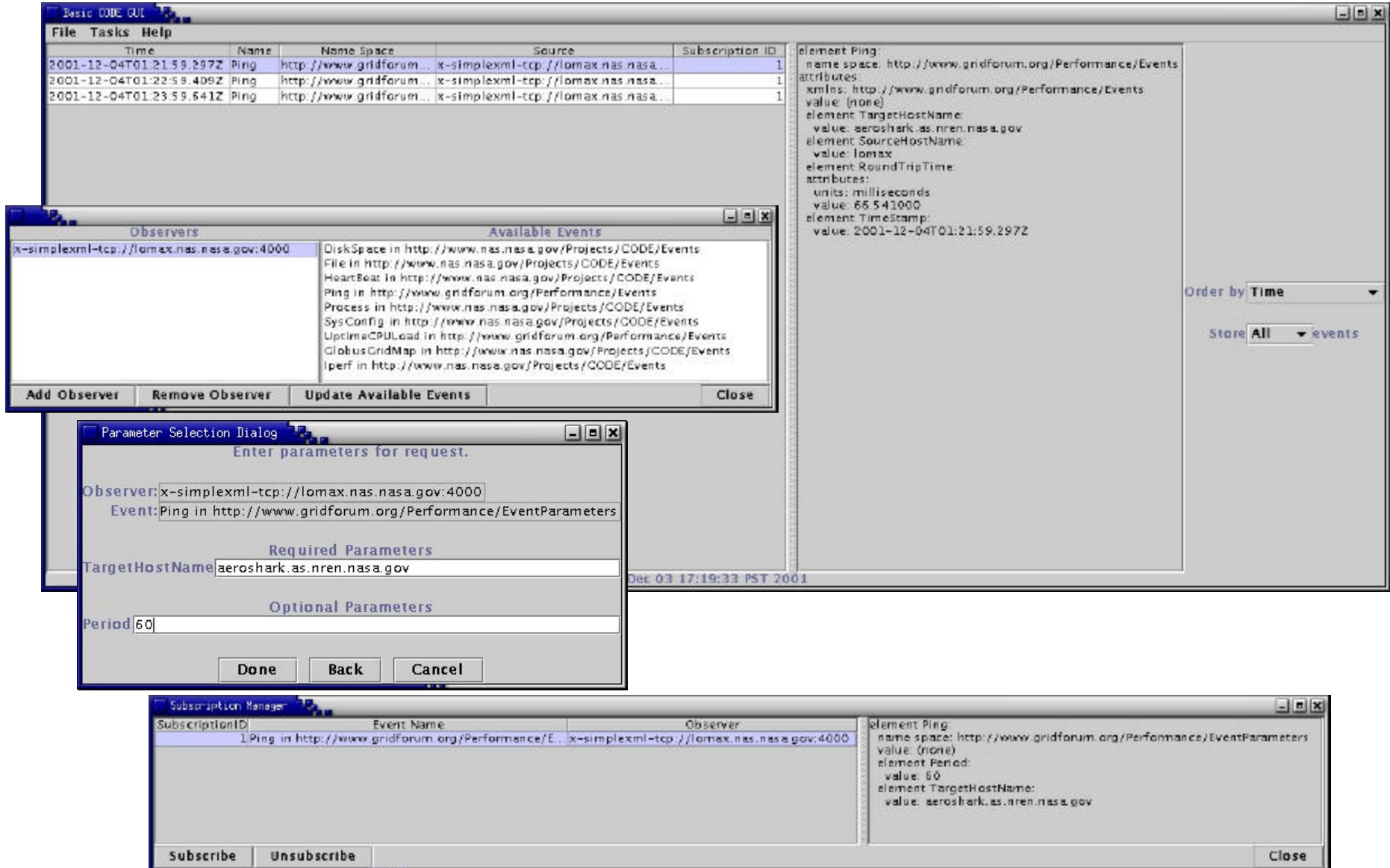
Grid Management System



Grid Management System



Basic GUI



The screenshot displays the Basic GUI interface with several overlapping windows:

- Basic CODE GUI:** The main window showing a table of subscriptions and a detailed view of a selected event.
- Observers:** A window listing available observers and events.
- Available Events:** A window listing various system events.
- Parameter Selection Dialog:** A dialog box for entering request parameters.
- Subscription Manager:** A window for managing subscriptions.

Basic CODE GUI Table:

Time	Name	Name Space	Source	Subscription ID
2001-12-04T01:21:59.297Z	Ping	http://www.gridforum.org/Performance/Events	x-simplexml-tcp://lomax.nas.nasa.gov:4000	1
2001-12-04T01:22:59.409Z	Ping	http://www.gridforum.org/Performance/Events	x-simplexml-tcp://lomax.nas.nasa.gov:4000	1
2001-12-04T01:23:59.641Z	Ping	http://www.gridforum.org/Performance/Events	x-simplexml-tcp://lomax.nas.nasa.gov:4000	1

Parameter Selection Dialog:

Enter parameters for request.

Observer: x-simplexml-tcp://lomax.nas.nasa.gov:4000

Event: Ping in http://www.gridforum.org/Performance/EventParameters

Required Parameters

TargetHostName: aeroshark.as.nren.nasa.gov

Optional Parameters

Period: 60

Buttons: Done, Back, Cancel

Subscription Manager Table:

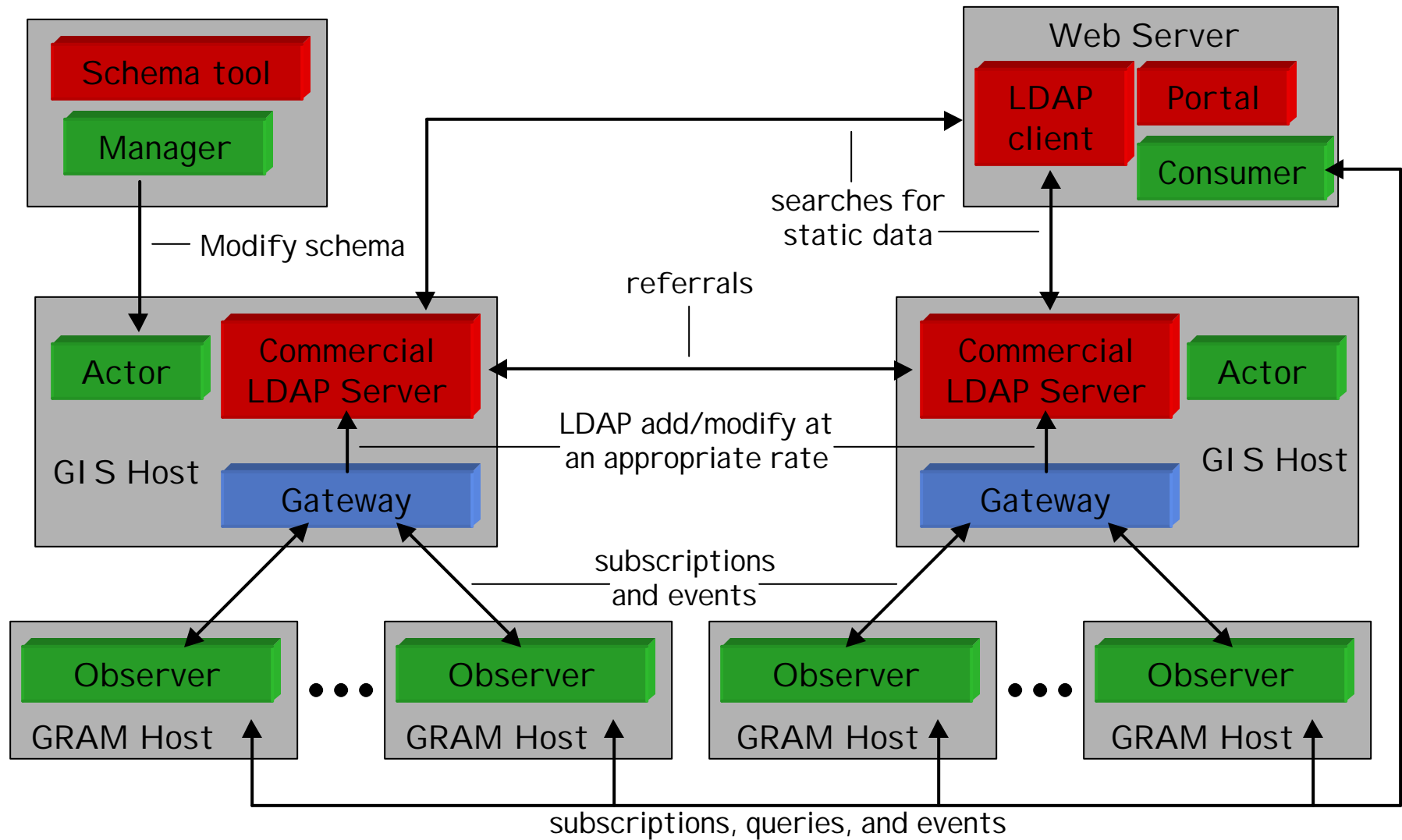
SubscriptionID	Event Name	Observer
1	Ping in http://www.gridforum.org/Performance/EventParameters	x-simplexml-tcp://lomax.nas.nasa.gov:4000

Buttons: Subscribe, Unsubscribe, Close

An Alternative GIS

- Explore different ways to implement information services
 - Take advantage of high-performance commercial LDAP servers
 - Satisfy requirements of our users
 - ◆ Modification of schemas by users
 - ◆ User areas
 - ◆ Availability: no root server, replicate top of LDAP hierarchy at each site
 - Build using CODE framework
 - ◆ Observers to gather data
 - ◆ Actors to modify LDAP server operation
 - Compare to Globus MDS
 - Deploy on I PG?
-

Alternative GIS Architecture



Alternative GIS Status

- Initial design complete
- Basic observation and control functionality ready
- Starting to implement now
 - ◆ Event-to-LDAP gateway
 - ◆ Sensors for exact LDAP entries
 - What LDAP information do we want?
 - Start with MDS 2 (not much from Grid Forum yet)
 - ◆ LDAP server management
 - ◆ GSI SASL in IPlanet servers (steal from Globus)
- First version, hopefully, in 2-3 months

Status and Future Work

- Current Status:
 - ◆ Available very soon to U.S. on a per-request basis
 - Public release in several months???
 - ◆ Code is beta quality, some missing functionality in Java
 - ◆ Preliminary grid management system
- Our future plans include:
 - ◆ Turn observers and actors into web services (convert protocols to SOAP/HTTP with WSDL defs)
 - ◆ More use of LDAP
 - ◆ Complete Java version (UDP and SSL)
 - ◆ More ports of C++ code (UNIX OS, ...)
 - ◆ Develop more sensors and actuators
 - ◆ More testing and documentation
 - ◆ A public release!
 - ◆ Use the framework to build tools
 - ◆ Stay compatible with Grid Forum standards